In [2]:

```
import pandas as pd
```

In [19]:

```
import matplotlib
```

In [4]:

```
from matplotlib import pyplot as plt
```

In [3]:

```
fb_data=pd.read_csv('Euro_2012_stats_TEAM.csv')
fb_data
```

Out[3]:

Team	Goals	Shots on target	Shots off target	Shooting Accuracy	% Goals-to-shots	Total shots (inc. Blocked)	Hit Woodwork	Penalty goals	Penalties not scored	...	S r
Croatia	4	13	12	51.9%	16.0%	32	0	0	0	...	
Czech Republic	4	13	18	41.9%	12.9%	39	0	0	0	...	
Denmark	4	10	10	50.0%	20.0%	27	1	0	0	...	
England	5	11	18	50.0%	17.2%	40	0	0	0	...	
France	3	22	24	37.9%	6.5%	65	1	0	0	...	
Germany	10	32	32	47.8%	15.6%	80	2	1	0	...	
Greece	5	8	18	30.7%	19.2%	32	1	1	1	...	
Italy	6	34	45	43.0%	7.5%	110	2	0	0	...	
therlands	2	12	36	25.0%	4.1%	60	2	0	0	...	
Poland	2	15	23	39.4%	5.2%	48	0	0	0	...	
Portugal	6	22	42	34.3%	9.3%	82	6	0	0	...	
epublic of Ireland	1	7	12	36.8%	5.2%	28	0	0	0	...	
Russia	5	9	31	22.5%	12.5%	59	2	0	0	...	
Spain	12	42	33	55.9%	16.0%	100	0	1	0	...	
Sweden	5	17	19	47.2%	13.8%	39	3	0	0	...	
Ukraine	2	7	26	21.2%	6.0%	38	0	0	0	...	

; × 35 columns

In [23]:

```
fb_data.describe()
```

Out[23]:

	Goals	Shots on target	Shots off target	Total shots (inc. Blocked)	Hit Woodwork	Penalty goals	Penalties not scored	Headed goals
count	16.000000	16.000000	16.000000	16.000000	16.000000	16.000000	16.0000	16.000000
mean	4.750000	17.125000	24.937500	54.937500	1.250000	0.187500	0.0625	1.375000
std	2.886751	10.582218	10.680005	26.065223	1.612452	0.403113	0.2500	1.024695
min	1.000000	7.000000	10.000000	27.000000	0.000000	0.000000	0.0000	0.000000
25%	2.750000	9.750000	18.000000	36.500000	0.000000	0.000000	0.0000	0.750000
50%	4.500000	13.000000	23.500000	44.000000	1.000000	0.000000	0.0000	1.500000
75%	5.250000	22.000000	32.250000	68.750000	2.000000	0.000000	0.0000	2.000000
max	12.000000	42.000000	45.000000	110.000000	6.000000	1.000000	1.0000	3.000000

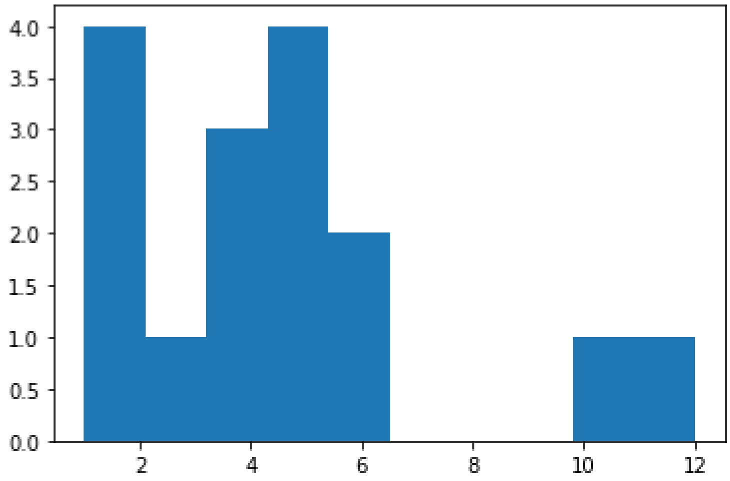
8 rows × 30 columns

In [24]:

```
plt.hist(x='Goals',data=fb_data)
```

Out[24]:

```
(array([4., 1., 3., 4., 2., 0., 0., 0., 1., 1.]),
 array([ 1. ,  2.1,  3.2,  4.3,  5.4,  6.5,  7.6,  8.7,  9.8, 10.9, 12. ]),
 <BarContainer object of 10 artists>)
```



In [5]:

```
fb_data['Penalty goals'].value_counts()
```

Out[5]:

```
0    13
1     3
Name: Penalty goals, dtype: int64
```

In [26]:

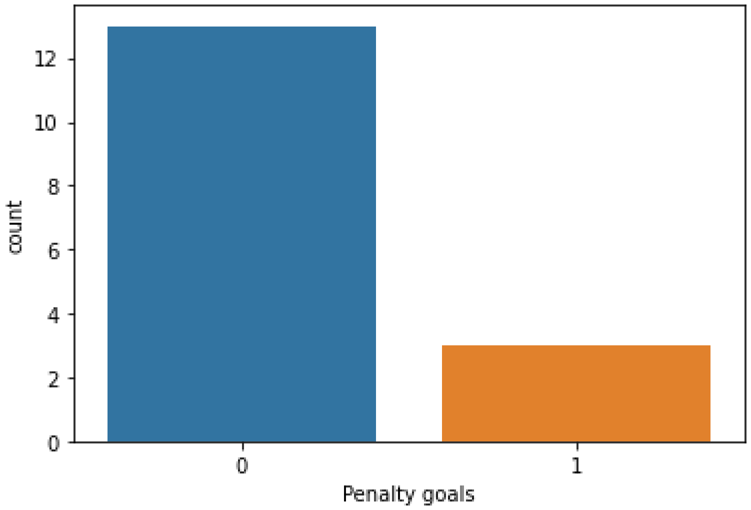
```
import seaborn as sns
```

In [27]:

```
sns.countplot(x='Penalty goals',y=None,data=fb_data)
```

Out[27]:

```
<AxesSubplot:xlabel='Penalty goals', ylabel='count'>
```



In [28]:

```
fb_data['Red Cards'].unique
```

Out[28]:

```
<bound method Series.unique of 0     0
1     0
2     0
3     0
4     0
5     0
6     1
7     0
8     0
9     1
10    0
11    1
12    0
13    0
14    0
15    0
Name: Red Cards, dtype: int64>
```

In [29]:

```
fb_data['Red Cards'].value_counts()
```

Out[29]:

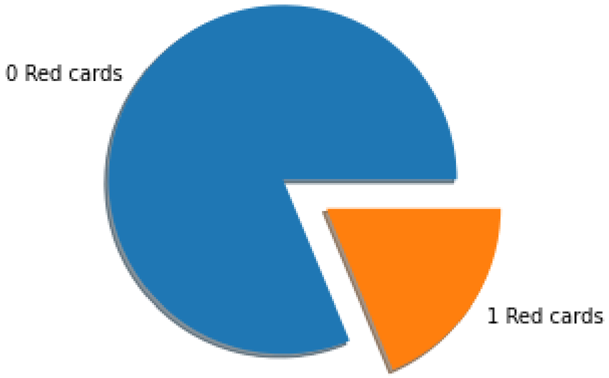
```
0    13
1     3
Name: Red Cards, dtype: int64
```

In [30]:

```
plt.pie(x=fb_data['Red Cards'].value_counts(),shadow=True,explode=[0.1,0.2],labels=['0 Red
```

Out[30]:

```
([<matplotlib.patches.Wedge at 0x230bac38fd0>,
  <matplotlib.patches.Wedge at 0x230bac45760>],
 [Text(-0.9977635347630545, 0.6666842796235226, '0 Red cards'),
  Text(1.080910495993309, -0.7222413029254828, '1 Red cards')])
```



In [31]:

```
round(fb_data.groupby('Penalty goals')['Goals'].mean())
```

Out[31]:

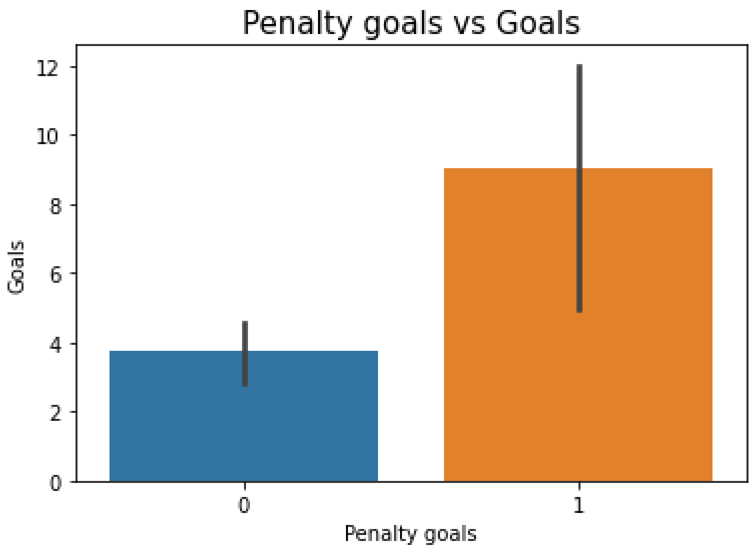
```
Penalty goals
0    4.0
1    9.0
Name: Goals, dtype: float64
```

In [32]:

```
sns.barplot(x='Penalty goals',y='Goals',data=fb_data,)
plt.title('Penalty goals vs Goals',size=15)
```

Out[32]:

```
Text(0.5, 1.0, 'Penalty goals vs Goals')
```

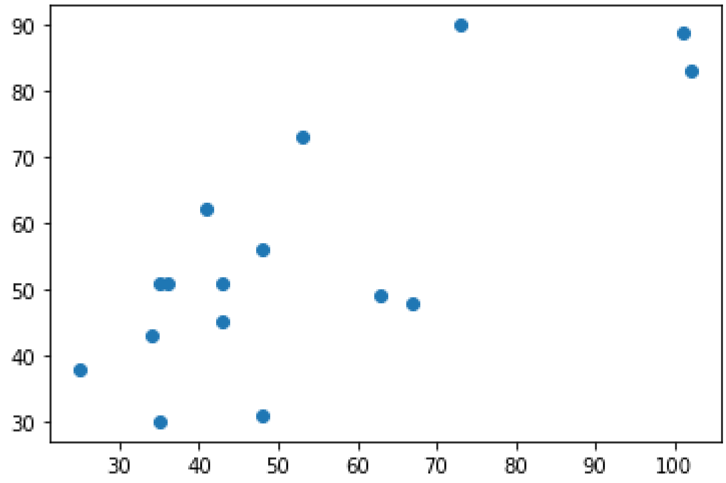


In [33]:

```
plt.scatter(x='Fouls Won',y='Fouls Conceded',data=fb_data)
```

Out[33]:

```
<matplotlib.collections.PathCollection at 0x230bace8a90>
```

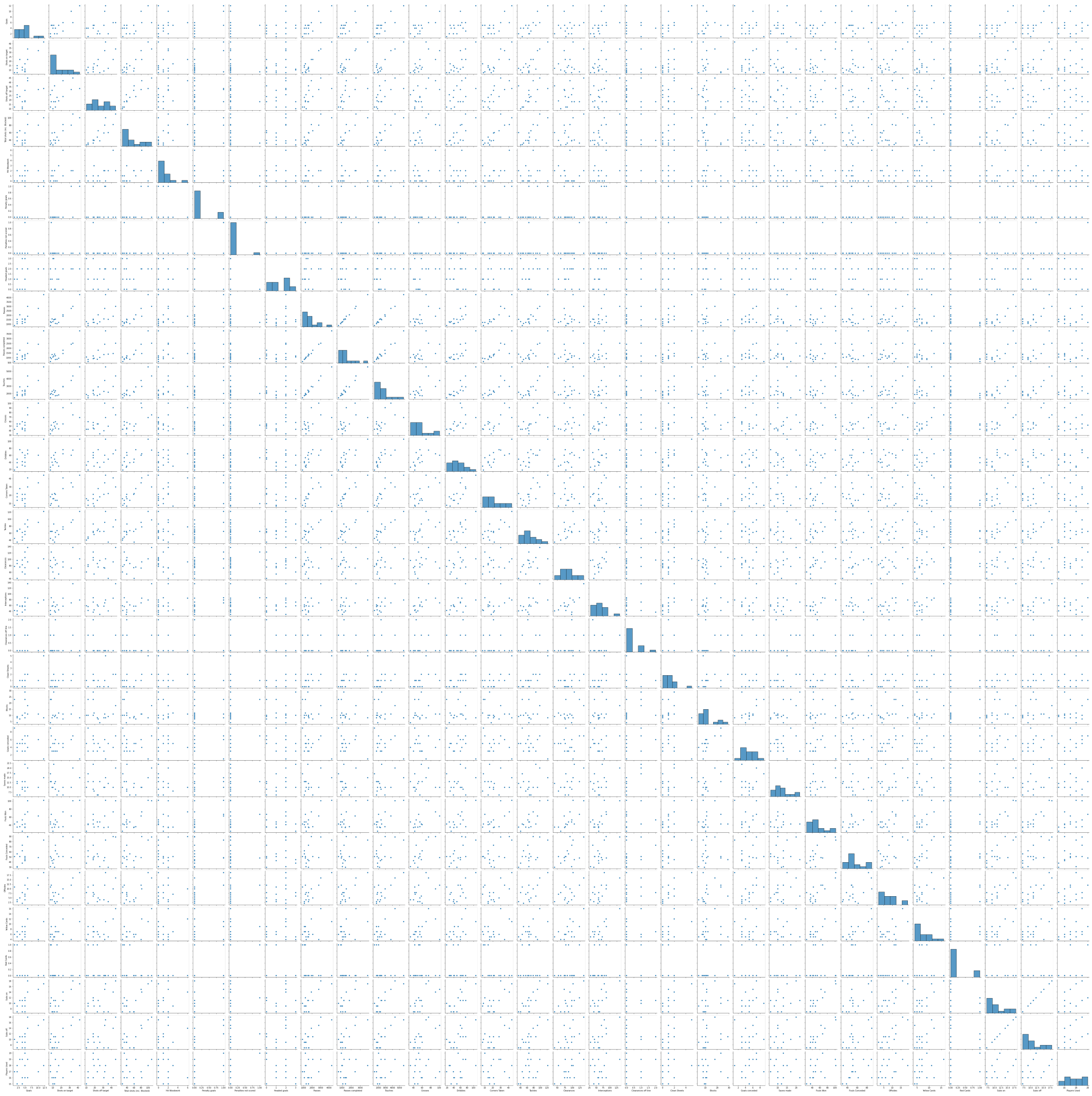


In [34]:

```
sns.pairplot(fb_data)
```

Out[34]:

```
<seaborn.axisgrid.PairGrid at 0x230bbca2eb0>
```



correlation matrix

In [6]:

```
corr_matrix=fb_data.corr()
corr_matrix
```

Out[6]:

	Goals	Shots on target	Shots off target	Total shots (inc. Blocked)	Hit Woodwork	Penalty goals	Penalties not scored
Goals	1.000000	0.804193	0.403821	0.650109	0.200512	0.730437	0.023094
Shots on target	0.804193	1.000000	0.611185	0.886090	0.212932	0.478610	-0.229945
Shots off target	0.403821	0.611185	1.000000	0.887032	0.581655	0.126783	-0.173221
Total shots (inc. Blocked)	0.650109	0.886090	0.887032	1.000000	0.400121	0.299397	-0.234668
Hit Woodwork	0.200512	0.212932	0.581655	0.400121	1.000000	-0.076923	-0.041345
Penalty goals	0.730437	0.478610	0.126783	0.299397	-0.076923	1.000000	0.537484
Penalties not scored	0.023094	-0.229945	-0.173221	-0.234668	-0.041345	0.537484	1.000000
Headed goals	0.349330	0.210571	0.002284	0.125738	0.020174	-0.020174	-0.357830
Passes	0.838497	0.930337	0.624451	0.875698	0.095882	0.545294	-0.169613
Passes completed	0.830062	0.923242	0.612998	0.864279	0.073814	0.550720	-0.179278
Touches	0.837760	0.934623	0.649127	0.894708	0.129331	0.534153	-0.133834
Crosses	0.667097	0.727451	0.563719	0.691374	0.480272	0.448069	-0.074308
Dribbles	0.693998	0.787066	0.510942	0.744255	0.061463	0.452092	0.008075
Corners Taken	0.697240	0.803347	0.749154	0.864856	0.389104	0.357828	-0.274399
Tackles	0.796830	0.811582	0.531234	0.757590	0.037136	0.528052	-0.039433
Clearances	0.255700	0.262694	0.212638	0.306509	-0.173871	0.261227	0.396836
Interceptions	0.482441	0.587278	0.531190	0.669220	0.307694	0.284426	0.242572
Clearances off line	-0.155292	-0.095605	-0.199107	-0.116198	-0.330719	-0.279508	-0.149404
Clean Sheets	0.792559	0.756715	0.348872	0.619025	0.088181	0.489003	-0.012926
Blocks	-0.029128	-0.172650	-0.230736	-0.180443	-0.102803	0.102803	0.401200
Goals conceded	-0.337689	-0.214415	-0.139274	-0.184740	0.062541	-0.020847	0.302532

	Goals	Shots on target	Shots off target	Total shots (inc. Blocked)	Hit Woodwork	Penalty goals	Penalties not scored
Saves made	0.143912	0.085059	0.086157	0.148269	-0.230524	0.058761	0.051019
Fouls Won	0.663876	0.765949	0.615058	0.766215	0.141180	0.528185	0.163643
Fouls Conceded	0.483833	0.658353	0.434815	0.623288	0.303987	0.114404	-0.107169
Offsides	0.703043	0.696263	0.344626	0.589635	0.113657	0.627220	0.210417
Yellow Cards	0.337709	0.544307	0.534186	0.622268	0.319708	0.085467	0.127603
Red Cards	-0.358057	-0.334050	-0.337766	-0.360466	-0.282051	0.179487	0.537484
Subs on	0.727830	0.821342	0.544974	0.763336	0.204511	0.531727	0.084796
Subs off	0.727830	0.821342	0.544974	0.763336	0.204511	0.531727	0.084796
Players Used	0.136067	0.249516	0.045973	0.157812	-0.189466	0.351866	0.480079

30 rows × 30 columns

In [66]:

```
fb_data.head()
```

Out[66]:

	Team	Shots on target	Shots off target	Shooting Accuracy	% Goals-to-shots	Total shots (inc. Blocked)	Hit Woodwork	Penalty goals	Penalties not scored	Headed goals
0	Croatia	13	12	51.9%	16.0%	32	0	0	0	2
1	Czech Republic	13	18	41.9%	12.9%	39	0	0	0	0
2	Denmark	10	10	50.0%	20.0%	27	1	0	0	3
3	England	11	18	50.0%	17.2%	40	0	0	0	3
4	France	22	24	37.9%	6.5%	65	1	0	0	0

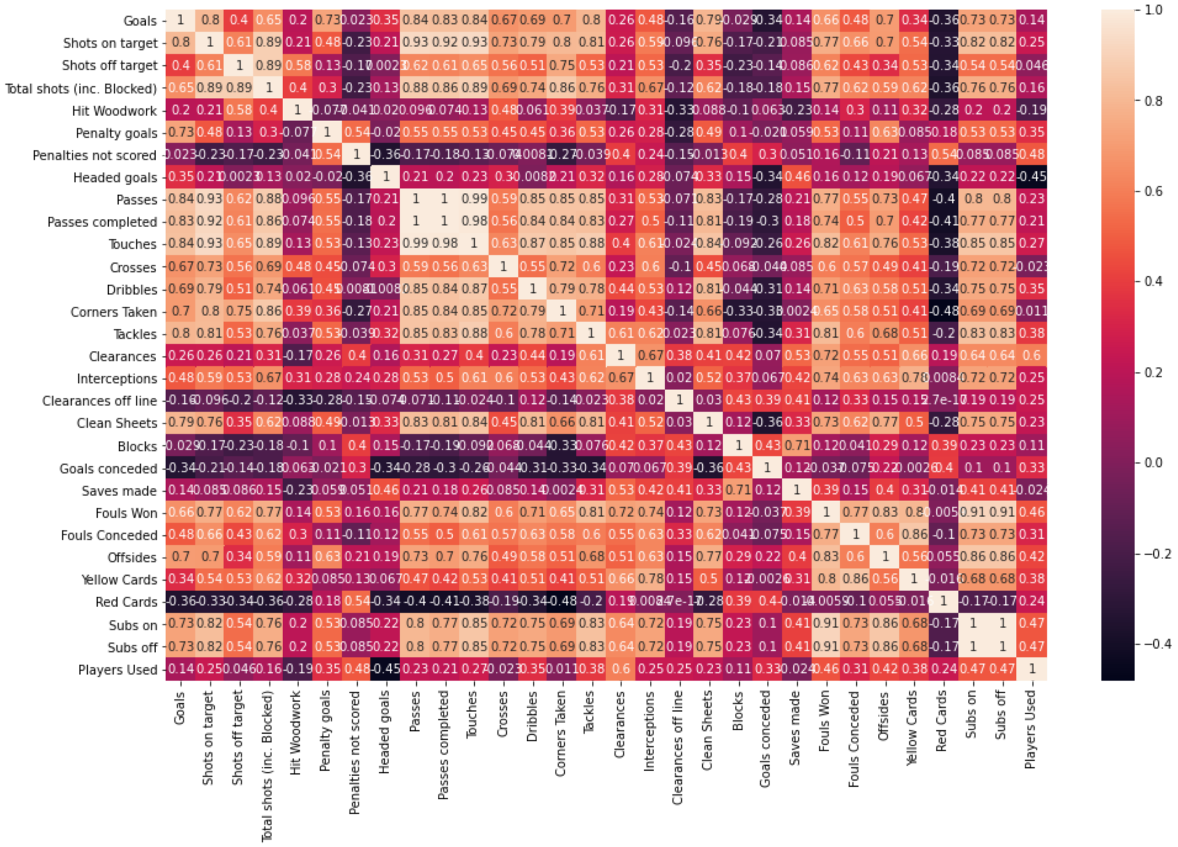
5 rows × 33 columns

In [10]:

```
import seaborn as sns
plt.figure(figsize=(16,10))
sns.heatmap(data=corr_matrix, annot=True,)
```

Out[10]:

```
<AxesSubplot:>
```



In [55]:

```
fb_data.describe()
```

Out[55]:

	Goals	Shots on target	Shots off target	Total shots (inc. Blocked)	Hit Woodwork	Penalty goals	Penalties not scored	Headed goals
count	16.000000	16.000000	16.000000	16.000000	16.000000	16.000000	16.0000	16.000000
mean	4.750000	17.125000	24.937500	54.937500	1.250000	0.187500	0.0625	1.375000
std	2.886751	10.582218	10.680005	26.065223	1.612452	0.403113	0.2500	1.024695
min	1.000000	7.000000	10.000000	27.000000	0.000000	0.000000	0.0000	0.000000
25%	2.750000	9.750000	18.000000	36.500000	0.000000	0.000000	0.0000	0.750000
50%	4.500000	13.000000	23.500000	44.000000	1.000000	0.000000	0.0000	1.500000
75%	5.250000	22.000000	32.250000	68.750000	2.000000	0.000000	0.0000	2.000000
max	12.000000	42.000000	45.000000	110.000000	6.000000	1.000000	1.0000	3.000000

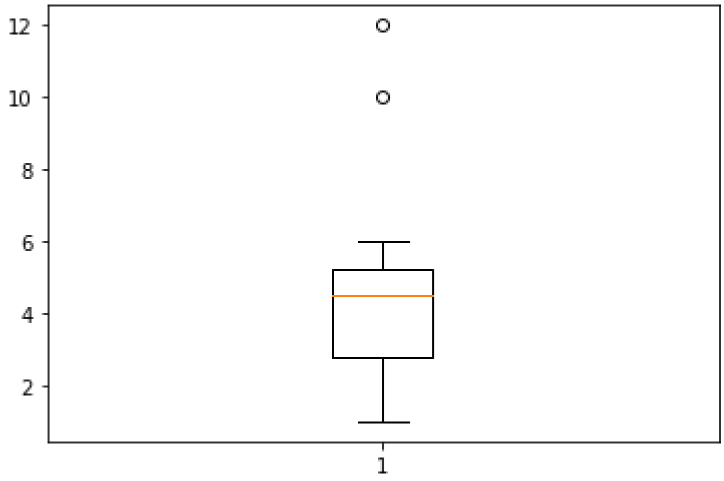
8 rows × 30 columns

In [54]:

```
plt.boxplot(x='Goals', data=fb_data,)
```

Out[54]:

```
{'whiskers': [<matplotlib.lines.Line2D at 0x230ea7a0cd0>,
  <matplotlib.lines.Line2D at 0x230ea7a22e0>],
 'caps': [<matplotlib.lines.Line2D at 0x230ea7a2700>,
  <matplotlib.lines.Line2D at 0x230ea7a2d60>],
 'boxes': [<matplotlib.lines.Line2D at 0x230ea7a0370>],
 'medians': [<matplotlib.lines.Line2D at 0x230ea4cb280>],
 'fliers': [<matplotlib.lines.Line2D at 0x230ea7a3130>],
 'means': []}
```

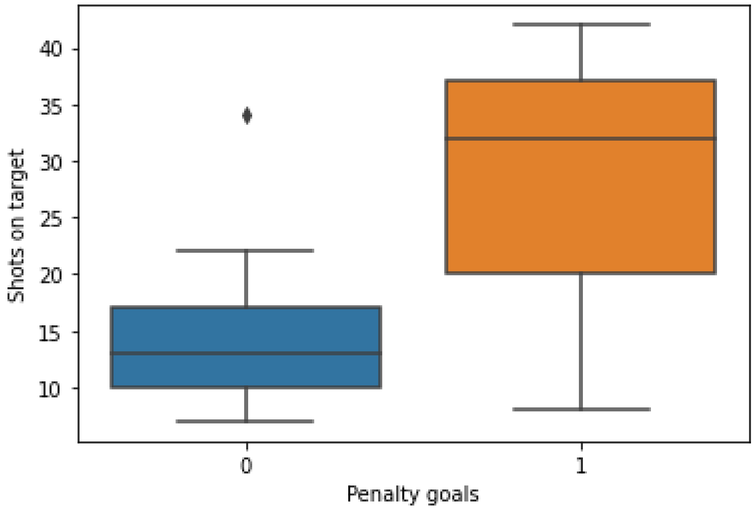


In [83]:

```
sns.boxplot( x='Penalty goals',y='Shots on target',data=fb_data,)
```

Out[83]:

```
<AxesSubplot:xlabel='Penalty goals', ylabel='Shots on target'>
```



In [ ]:

```

```